MINT
version 1.0
Users' Manual

Anna Górska
Maciej Jasiński
Joanna Trylska

Contents

1 Introduction

MINT stands for Motif Identifier for Nucleic acids Trajectory - this is an automatic tool for reading and analyzing RNA three-dimensional structures and molecular dynamics trajectories (written in CHARMM, Gromacs, NAMD, LAMMPS, or Amber formats). The analysis includes RNA hydrogen network pattern and structural motifs. These properties can be investigated as a function of the conformations provided by the user. In the case of molecular dynamics trajectory MINT gives the flow of various RNA properties in time.

Please direct your comments and questions about the MINT package to:
agorska@cent.uw.edu.pl
Centre of New Technologies, University of Warsaw
Żwirki i Wigury 93, 02-089 Warsaw, Poland
phone: +48 22 5540 843

The web interface to analyze RNA structures is available at:

https://bionano.cent.uw.edu.pl/MINT

and the source code of the MINT package to analyze multiple snapshots, for example, molecular dynamics trajectories can be found at:

https://bionano.cent.uw.edu.pl/Software

If you find this software useful please cite: Anna Górska, Maciej Jasiński, Joanna Trylska, TITLE, JOURNAL, submitted

2 Quick reference

2.1 What our software can do?

For a single .pdb file of an RNA molecule MINT computes and outputs:

- regions forming helices,
- regions forming loops, bulges, interior loops, junctions,
- regions forming pseudo-knots,
- nucleotides creating triplexes,
- all Watson-Crick base pairs,
- all non-Watson-Crick base pairs,

- the number of Watson-Crick hydrogen bonds per nucleotide,
- the number of non-Watson Crick hydrogen bonds per nucleotide,
- the stacking energy – Van der Waals and electrostatic energies and their sum per nucleotide,
- visualizations of the computed parameters.

For a trajectory file it computes and outputs:

- residues forming helices, triplexes, pseudo-knots and other motifs, as well as frame numbers and percentage of time when they were present,
- clusters of secondary structure motifs and average motifs along with two-dimensional and three-dimensional contacts,
- all Watson-Crick and non-Watson-Crick base pairs and frame numbers in which a helix was present as well as percentage of trajectory time the helix occurred,
- the average number of Watson-Crick and non-Watson Crick hydrogen bonds per nucleotide,
- the average stacking energy – Van der Waals and electrostatic energies and their sum per nucleotide,
- average secondary structure,
- visualizations of the computed parameters (see section 7.1)

2.2 What does it need?

- Python2.7 and described below packages.
- an RNA structure file in a full-atom (including hydrogens) representation and in the .pdb format.
- a trajectory file (e.g. in the .dcd format) in case you want to use the `Traj` mode.

2.3 Example

On the website we provide an example.tar package containing **inputs**:

- `example.pdb` – a pdb file containing the atomic structure of the RNA molecule. One will benefit from the program mostly while analyzing the RNA structures with a complex secondary an tertiary structures.
- `example.dcd` – a trajectory file containing ten frames from molecular dynamics simulations performed with NAMD [12] and using the CHARMM forcefield.

and **outputs**:

- Structure description (see section 7 for details):

  - example_description
  - example_RNA_ANALIZER.xls
  - example_nucleotides_eval.csv
  - example_pairs_in_time.csv
  - example_per_nucleotide.csv
  - example_average_motifs.csv
  - example_helices_in_time.csv
  - example_motifs_clusters.csv

- Visualization (see section 7.1 for details):
  - exampleRNAStructML.xml
  - example_varna.html
  - example_2D.pdb
  - example_3D.pdb
  - example_VDW.pdb
  - example_stacking_sum.pdb
  - example_coulomb.pdb
  - vmd_run.tcl

All the examples provided in this manual are based on the above example.

2.4 Running

After downloading and unpacking the package `MINT.tar`, you can go straight to the `MINT` directory and type

```
python MINT.py
```

and the program should perform a single frame analysis of the `example.pdb`. If any errors appear check whether you are using python2.7 and have all the required packages installed (the packages are listed in the section 3).

If you would like to perform single frame analysis of your input pdb with hydrogens

If you want to perform the trajectory analysis of the `example.dcd` file, type:

```
python MINT.py Single/Traj=Single
```

Running program with the `help` option will print all the parameter names and their default values:

```
python MINT.py help
```

To set your own parameters simply type the name of the parameter "=" value list separated with spaces:

```
python MINT.py Single/Traj=Traj vmd=0 h\_bond\_angle=130 cutoff=30 threads=1
```

Another way is to directly edit the `parms` dictionary in the end of the `pairs.py` file in the main function. You can use any text editor but watch out for quotation marks, indents and commas.

3 External modules and installation

The script uses several external python packages:

- python 2.7 – Python programming language (`http://python.org`)
- numpy – the package containing base tools to manipulate multi-dimensional arrays, the installation is described on the SciPy home website: `http://www.scipy.org/Download`
- BioPython – the main package that enables loading and managing the PDB structure, the installation is described on the BioPython home website: `http://biopython.org/wiki/Main_Page`
- MDAnalysis [11] – used only for reading the trajectory, so as long as a single .pdb file is analyzed, the `import MDAnalysis` line can be removed. The instruction how to install this Python package can be found here `http://code.google.com/p/mdanalysis/wiki/Install` - MDAnalysis home website.
- xlwt – the package for writing .xls files. It is used by `csvToxls.py` script to write all output .csv files into the .xls files (can be installed from pypi website `https://pypi.python.org/pypi/xlwt`).
- pympler – the package enabling measuring the memory use of the objects in python script - can be downloaded from its website `http://pythonhosted.org/Pympler/`.

all of them can be installed through `easy_install` command available in `setup_tools` package (`http://pythonhosted.org/distribute/easy_install.html#installing-easy-install`.

multipy If you encounter any kind of problems with installing python2.7 or required packages, you can use a Multipy package. This package allows to set up a local virtual python environment and run the script. The Multipy can be downloaded from the website `http://code.google.com/p/multipy/`. To install python 2.7 type

```
multipy install 2.7
```

then you can access the python 2.7 multipy environment by using:

```
bash
. $(multipy activate 2.7)
```

and now it is easy to install the needed packages:

```
easy_install numpy
easy_install Biopython
easy_install MDAnalysis
easy_install xlwt
easy_install pympler
```

4 Definitions

Here we outline a few concepts/definitions that we use for the purpose of the manual and running software. The meaning of some has been be narrowed and of others extended in comparison to their biological meaning:

- Canonical pair – all Watson-Crick pairs, including AG, AC and others.
- Non-canonical pair – all non-Watson-Crick pairs.
- RNA secondary structure – created by canonical pairs excluding pseudo-knots.
- RNA tertiary structure – created only by non-canonical pairs.
- Motif – a loop, bulge or junction but not a helix.

5 Workflow

Figure 1 shows the structure of the program. First, the program reads in the RNA conformation and finds all hydrogen bonds between nucleotides. They allow determining secondary and tertiary structures and subsequent classification of the RNA secondary structural motifs. The main outcome is a set of statistical descriptors of the given RNA structure. In this document we describe the algorithm, inputs and outputs of the program.

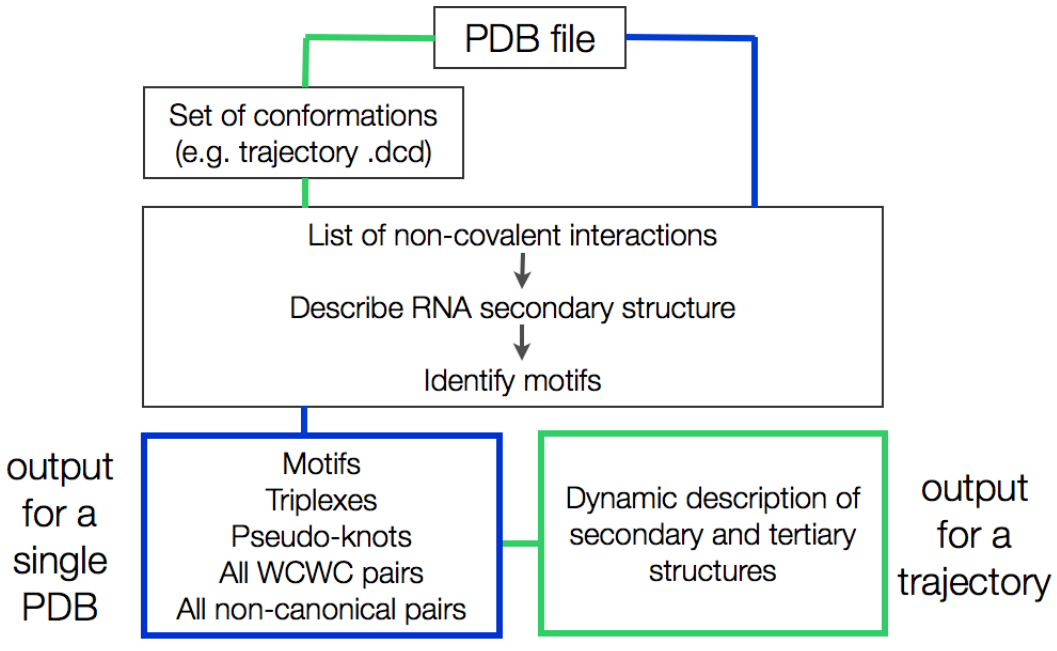


Figure 1: Main function implements the analysis of a single frame. In the case of a trajectory, the function reads-in the list of nucleotides from a given PDB and then refreshes the atom coordinates while reading the next frame. Additionally, the script splits the trajectory between CPUs and runs separate processes, what accelerates the calculations.

5.1 Loading the RNA structures

The script is entirely written in Python programming language (`http://python.org`). Its modular structure enables applying it in various programing contexts. Reading the PDB file is implemented in BioPython (`http://biopython.org/wiki/Main_Page`) package, providing a complete objective structure for dealing with PDBs. The basic object is an atom that despite the name and number is described with coordinates. A molecule consists of the residues that have such attributes as the name, number and list of atoms. This allows fast and easy access to the nucleotides, atoms and their coordinates.

5.2 Hydrogen bonding

A hydrogen bond is a basic interaction responsible for creating the RNA secondary and tertiary structures. A typical definition of a hydrogen bond pertains to a non-covalent interaction when a hydrogen atom is placed close to its acceptor.

Both the distance and the angle depend on the characteristics of the donor and acceptor (Figure 2). Theory states that all hydrogen bonds are almost linear (around 175°) [5]. For biological molecules the hydrogen bond distance should be typically between 2,80 and 3,06 Å between a donor and acceptor, which gives 1,60 and 1,80 Å between an acceptor and hydrogen.

The user defines both parameters: the minimal angle and maximal distance of a hydrogen bond. The default value of the h_bond distance is 2.8 Å and the minimal angle value is 140° (see Figure 2).

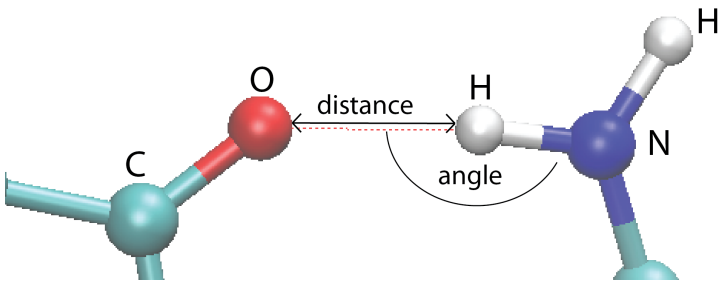


Figure 2: The scheme of the hydrogen bond with the nitrogen atom as a donor and the oxygen atom as an acceptor.

5.3 Donors and acceptors

To analyze the RNA structures we have defined a list of possible acceptors and donors for all four nucleotides (A, U, G, C). Following the classification by Leontis and Westhof [8] the acceptors and donors are assigned to the edges of the nucleotide, as shown in Figure 3. The program determines the interacting edges of every pair of nucleotides. Several atoms are situated in the corners of the nucleotides and participate in more than one edge. In that case, the program first classifies all the remaining bonds and chooses the prevailing edge. If there is only one hydrogen bond, both edge names are returned.

After determining the donors and acceptors of a molecule, as well as the hydrogen bond pattern, the program can now search for nucleotide pairs. The program has to check all the donors against

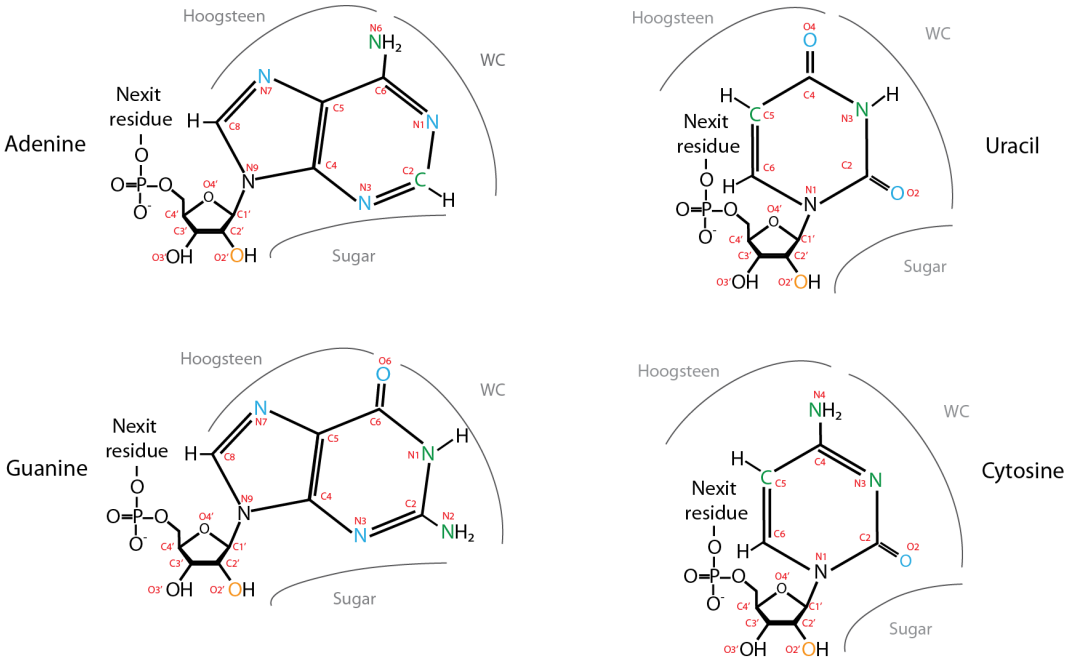
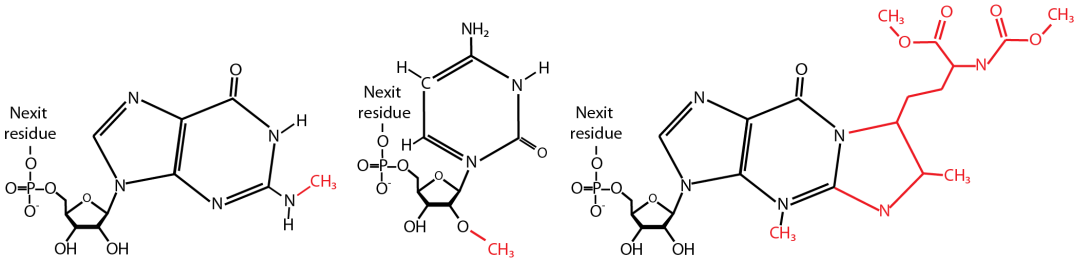


Figure 3: Nucleotides with selected edges, donors (green) and acceptors (blue). WC corresponds to Watson-Crick edge. [9].

all acceptors of all possible pairs of nucleotides in the RNA molecule. In order to decrease the computational time, we assumed that the partner for a given nucleotide can be found only among its closest neighbors - nucleotides placed in the space within a given `cutoff`. The exact distance is defined by the user (`cutoff`). Knowing the atoms participating in hydrogen bonds, the program can determine the interacting edges.

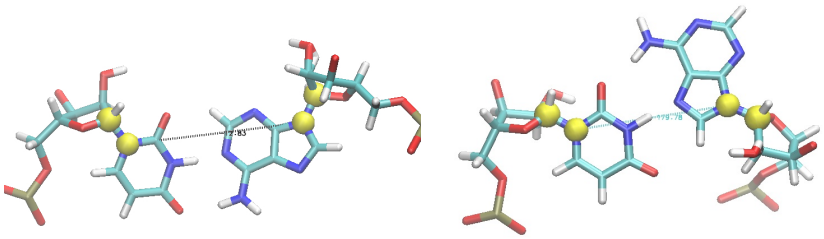
Modified nucleotides RNA structures sometimes include modified nucleotides such as 2N-methylguanosine-5'-monophosphate, 5,6-dihydrouridine-5'-monophosphate or N2-dimethylguanosine-5'-monophosphate that can be found, for example, in tRNA molecules. Several of these are presented in Figure 4. If you find modified nucleotides in the studied RNA system you should edit the `nucleotides.csv` file and add a new row with the name of your residue and appropriate names of atoms in the donors and acceptors columns. Than you should also modify the `charges_and_VDW` file for stacking computation. We could not have done it for you due to the not unified naming of the modified nucleotides and atoms they consist of.



(a) 2MG (b) OMC (c) YYG

Figure 4: Three out of ten modified nucleotides present in tRNA structure (PDB: 1ttt) along with their PDB ID. Red color indicates atoms that are not present in normal nucleotides.

Geometric isomerism After detecting the hydrogen bonds, and defining the interacting sites, the geometric isomerism is computed. The program measures the torsion angle formed by four atoms, and depending on its value the geometric isomerism is denoted. There are two possible isomerisms: cis and trans, both shown in Figure 5.



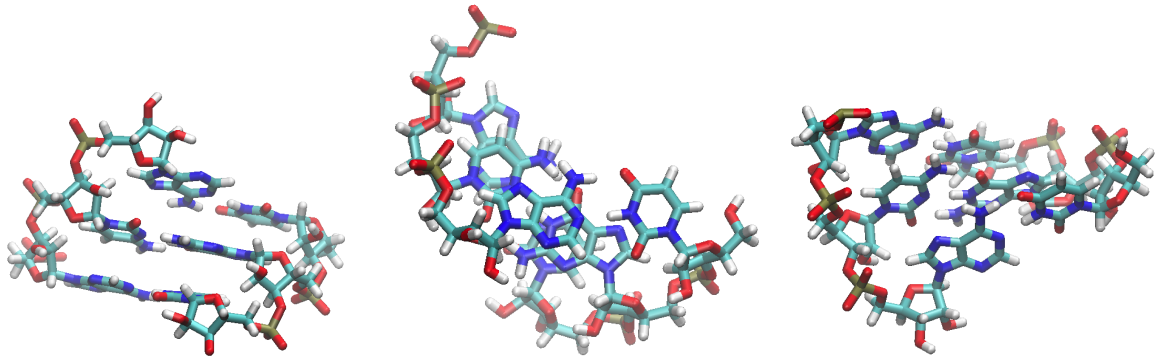
(a) Cis (b) Trans

Figure 5: Yellow spheres correspond to C1' and N1 atoms in pyrimidines and N9 atoms in purines. The torsion angle created by these four points determines the geometric isomerism of the two nucleotides creating a pair.

5.4 Stacking

Stacking is an important non-covalent interaction contributing to the stability of both double helix and single stranded structures of nucleic acids [6]. It is suspected to be especially important for RNA molecules. In tRNA only a half of the nucleotides form a helix, but about 90% of residues is stabilized by stacking [2].

Generally, stacking occurs between aromatic rings, in nucleic acids between the nucleobases. There is a general belief that the basis of stacking lies in the contact of the electron π-systems. Stacking is also supported by three phenomena: Van der Waals (dipole or induced-dipole attractions), electrostatic and solvation effects. It seems to be more important in folding of nucleic acids than proteins because nucleotide bases are more polarizable than most amino acids.



(a) (b) (c)

Figure 6: Three-pairs of an RNA helix seen from different angles to expose stacking between parallel bases.

We estimate stacking between two bases by calculating the energy of electrostatic (U_{el}) and Van der Waals (U_{VDW}) interactions applying the equations used in molecular mechanics:

$$U_{el} = k \sum \frac{q_i q_j}{r_{ij}} \tag{1}$$

$$U_{VDW} = 4\epsilon \sum \left[\frac{1}{4} \left(\frac{r_0}{r_{ij}} \right)^{12} - \frac{1}{2} \left(\frac{r_0}{r_{ij}} \right)^{6} \right] \tag{2}$$

The sums consist of all pairs containing atoms from nucleobases i and j. k denotes the Coulomb constant ($k = \frac{1}{4\Pi\epsilon_0}$), q is the partial atomic charge of the atom, r_{ij} is the distance between considered atoms, ϵ is the depth of the Lennard-Jones potential well for atoms i, j and r_0 is the sum of Van der Waals (VDW) radius of atoms i and j. We provide the VDW parameters and partial atomic charges from the Amber [**?**] and Charmm [**?**, **?**] force fields. A set of parameters and charges may be also defined by the user in the file `charges_andVDW.csv`. Only the nucleobases that are closer than the user defined cutoff (`cutoff_stacking`) are considered in the stacking calculations. The unit for energy values obtained from described calculations is $kcal/mol$.

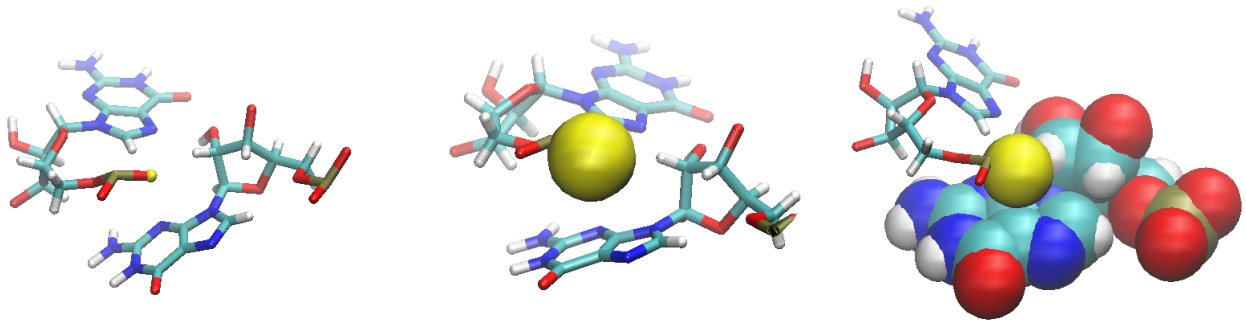
It is believed that stacking is one of the best represented terms in molecular modeling. Especially with the Amber atomic charges which are fitted to molecular electrostatic potentials [**?**]. It was shown that calculations using the empirical potentials consisting of the Lennard-Jones VDW and Coulombic terms with atom-centered point charges were able to reproduce the *ab initio* stacking energy over the major portion of the conformational space [10]. Šponer et al. in multiple works [**?**, **?**, **?**, **?**] compared *ab initio* energies for about 300 geometries of stacked base dimers with data obtained by using empirical potentials. The agreement between these methods is remarkable, which suggests that calculations based on empirical potentials provide an excellent approximation of the stacking interaction energy between nucleotides.

The electrostatic term, depending on the orientation of bases' dipole moments, may be attractive or repulsive regardless the bases are parallel or not to each other. While the VDW energy component is almost always favorable regardless base orientation. What is more, the shape of nucleotides and method used for calculation the VDW energy, ensure that the lowest VDW values

are obtained for parallel nucleotides which the biggest overlap. It was also visible in our test calculations. Thus we recognize two nucleotides as stacked, if their VDW energy is lower than `vdw_cutoff_stacking` parameter. The default value is $-0.5 kcal/mol$. It was found by trial and error and seems to be appropriate for the nonmodified nucleobases.

5.5 Ion-Π interaction

In biomolecules such as RNA, proteins and their complexes the non-covalent interactions play significant role. Hydrogen bonds are the most known non-covalent interactions but not the only one. For example the cation-Π interactions, namely the non-covalent bonding between a monopole (cation) and a quadrupole (Π system), seems to play important role in proteins structure. Similar interaction was reported for the RNA molecules, but it was found that nucleic acid aromatic systems prefer to interact with anionic rather than cationic species [**?**]. MINT allows to search for anion-Π interactions involving RNA backbone phosphate groups and nucleotides bases. The example is shown in Figure 7. Potentially interacting systems are recognized by distance criterium



(a) (b) (c)

Figure 7: Three representations of the oxygen atom (yellow) "stacking" over the guanine base. The yellow sphere in the picture corresponds to the real VDW radius.

and than energy of interaction between phosphate atom and nucleotide base is calculated in the same way as in the stacking.

5.6 Representation of the RNA motifs

Having all Watson-Crick pairs, a list-representation of the RNA secondary structure is created. We assume that one nucleotide can have only one Watson-Crick partner. If the second WCWC partner is encountered, all three nucleotides are denoted as a triplex. The index of the list represents the nucleotide number; the stored value is the index of its Watson-Crick partner. The list is easy-interpretable when the arcs connecting the pairs are drawn as presented in Figure 8.

Pseudo knots The list-representation contains also the information about the non-secondary motifs. The pseudo-knot is formed by the Watson-Crick interactions but creates three-dimensional folds as shown in Figure 9. Our program detects the pseudo-knot fold when the arcs intersect. A pseudo-knot is a symmetric structure – both the three pairs 6–17, 7–16, 8–15 in Figure 8 form the

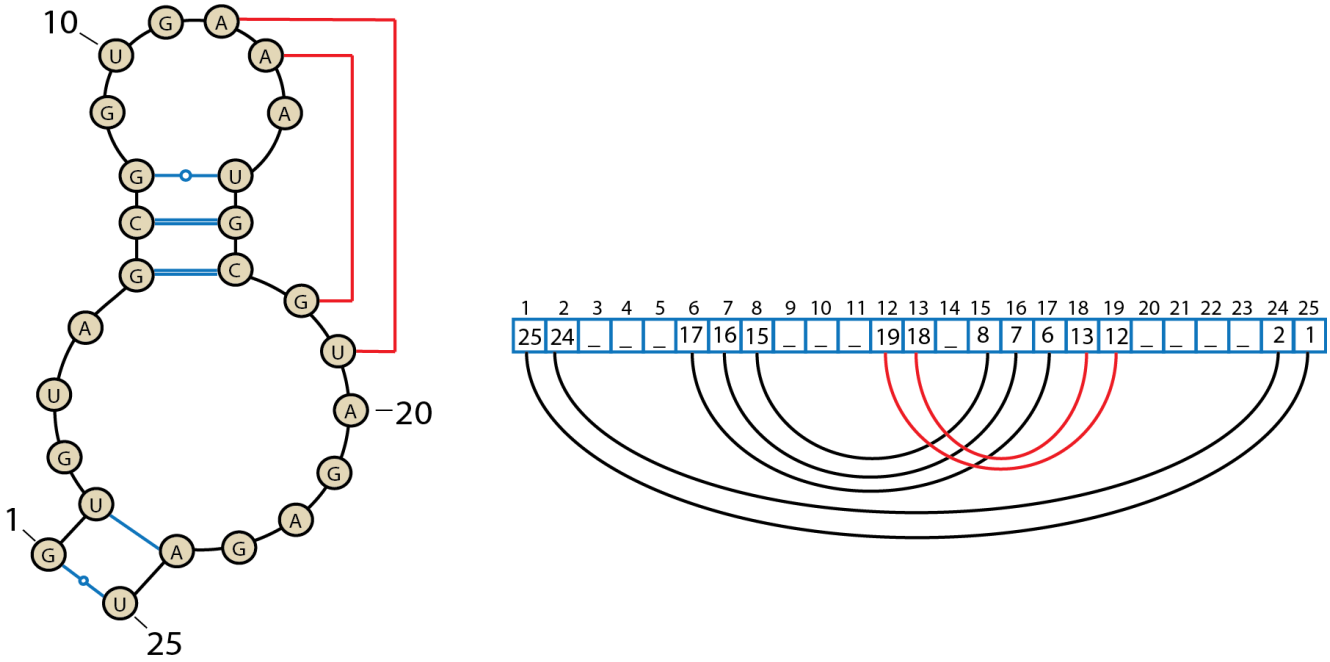
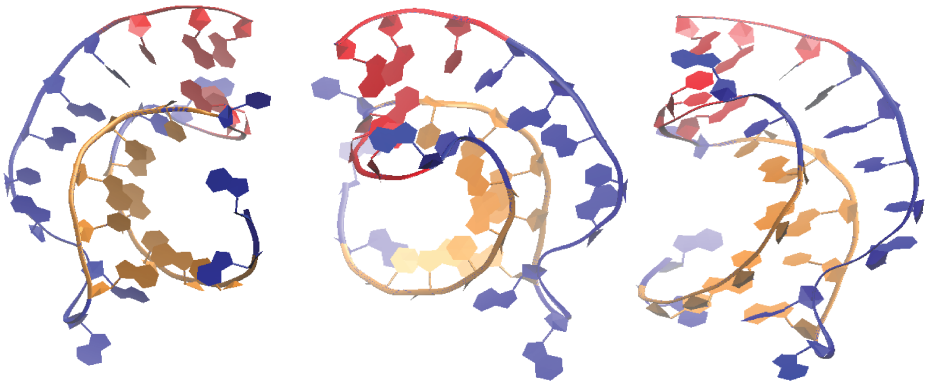
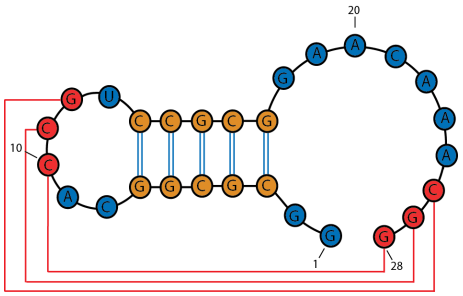


Figure 8: Secondary structure of an exemplary RNA molecule and its list-representation. Red lines correspond to the Watson-Crick interactions creating a pseudo-knot.

pseudo-knot as well as the two pairs: 12–19, 13–18. The natural way of solving this conflict is to choose the shorter list, in this case the pairs 12–19 and 13–18.



(a) Tertiary structure



(b) Secondary structure

Figure 9: An example of RNA structure with a pseudo-knot seen from three different angles. Nucleotides colored in red create a pseudo-knot, orange form a helix and blue refer to loops (PDB id: 437d).

After detecting all pairs, and creating the list, the program finds all pseudo-knots and erases them from the list by putting the `None` value. Next, the program finds all kinds of motifs, that are defined as a set of unpaired nucleotides with the surrounding pairs as it is show in the figure 10

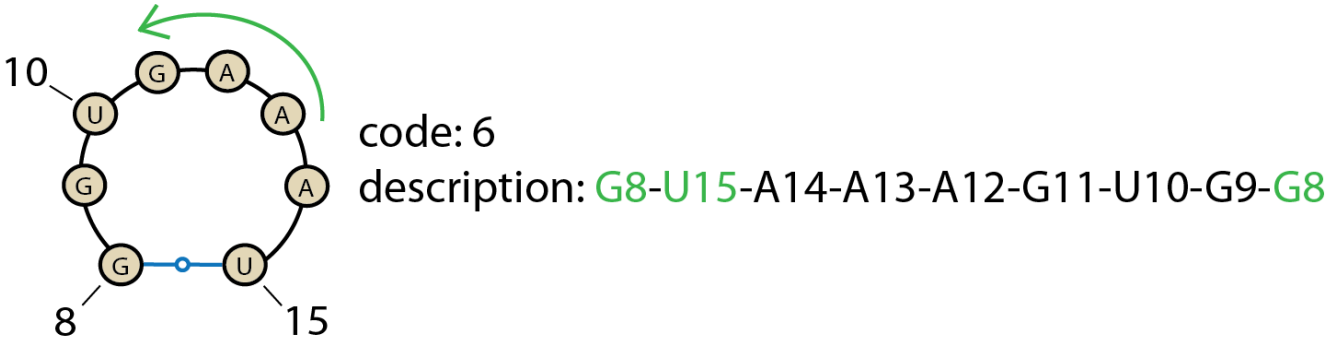
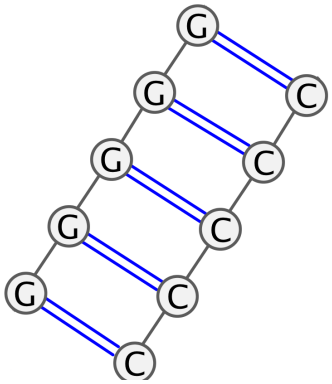
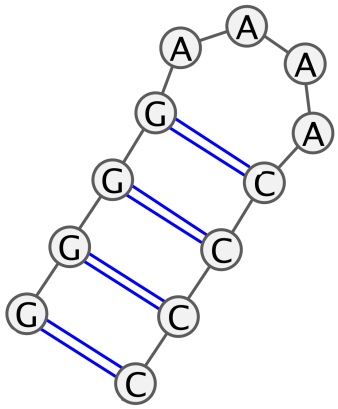


Figure 10: Program denotes the first pair (shown in green) and than list of unpaired nucleotides. Nucleotides are listed in accordance with the counter-clockwise direction.

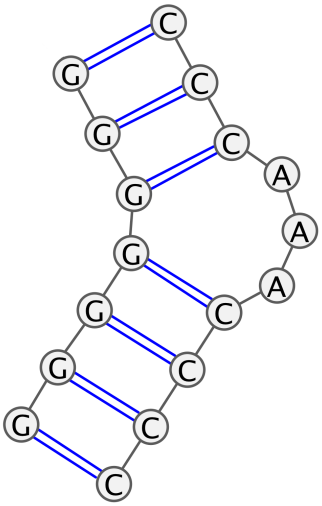
Table 1: RNA secondary structure motifs



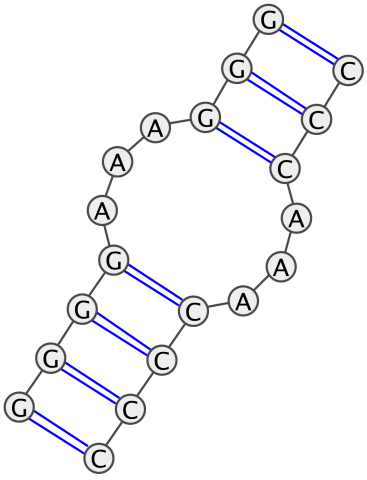
Helix



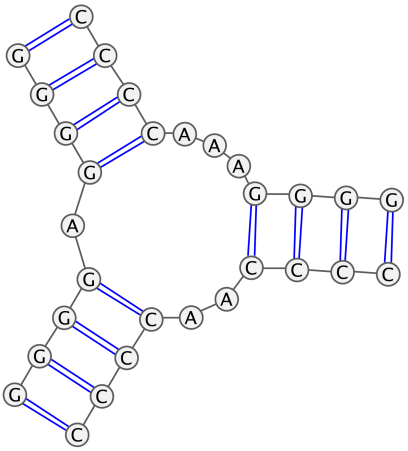
Hairpin loop
(code: 4)



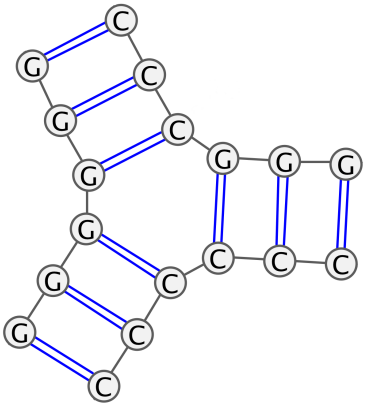
Bulge loop
(code: 0–3)



Interior loop
(code: 3–3)



Multi-branched loop
(code: 2–3–1)



3-way junction
(code: 0–0–0)

RNA motifs nomenclature (loop, bulge, junction) can be inaccurate and misleading. The program uses the numerical description of the motifs that is understandable for people. The numbers correspond to the numbers of unpaired nucleotides between pairs. A four nucleotide loop at the end of a hairpin is represented by a single number 4. A symmetric bulge loop, with both bulges of the length of three is represented by two numbers: 3–3 and a three-way junction, without unpaired nucleotides is represented by three zeros: 0–0–0. Different kinds of RNA secondary structure motifs are presented in Table 1.

5.7 Motif-search algorithm

The algorithm uses a list representation of the RNA-secondary structure that contains the list of Watson-Crick pairs. As shown in Figure 11 the algorithm detecting helices and other secondary structure motifs walks through the list and stores the information about the visited nucleotides. As long as there are paired nucleotides it stores the information about the helix. When it encounters the end of the helix – an unpaired nucleotide ahead of a pair, it starts to travel (search) around the motif. It stores the first pair as the beginning and goes to the index stored in the list. Then as long as there is an unpaired nucleotide it moves back – with decreasing indices. When another pair is encountered, the algorithm goes into the indicated position and again moves back. The motif ends when the algorithm finds itself one step ahead of the starting index. The number of "jumps" is the number of values and the values represent the number of unpaired nucleotides in between. After one motif is found and classified, the algorithm jumps one step further than the last seen pair. The algorithm stops searching for the motifs and helices when the index is larger than the value stored in the list.

As a result of the single frame analysis the program returns a list of all pairs, motifs and numerical characterization of all nucleotides. Every nucleotide is parametrized with the number of created hydrogen bonds and the description of its partner:

```
G538 , 538 , 3-C513A539:1 , 2-C513:2 , 3-C513:7
```

First the type of the nucleobase with its PDB id, nex the PDB id one more time, and lists pairs the nucleotide created during the trajectory. Description of the configuration starts with the number of hydrogen bonds, than the nucleobase and PDB ID. There can be more than one nucleotide listed - what indicated the triplex. The number after a semicolon is the number of frames this pair or triplex were present.

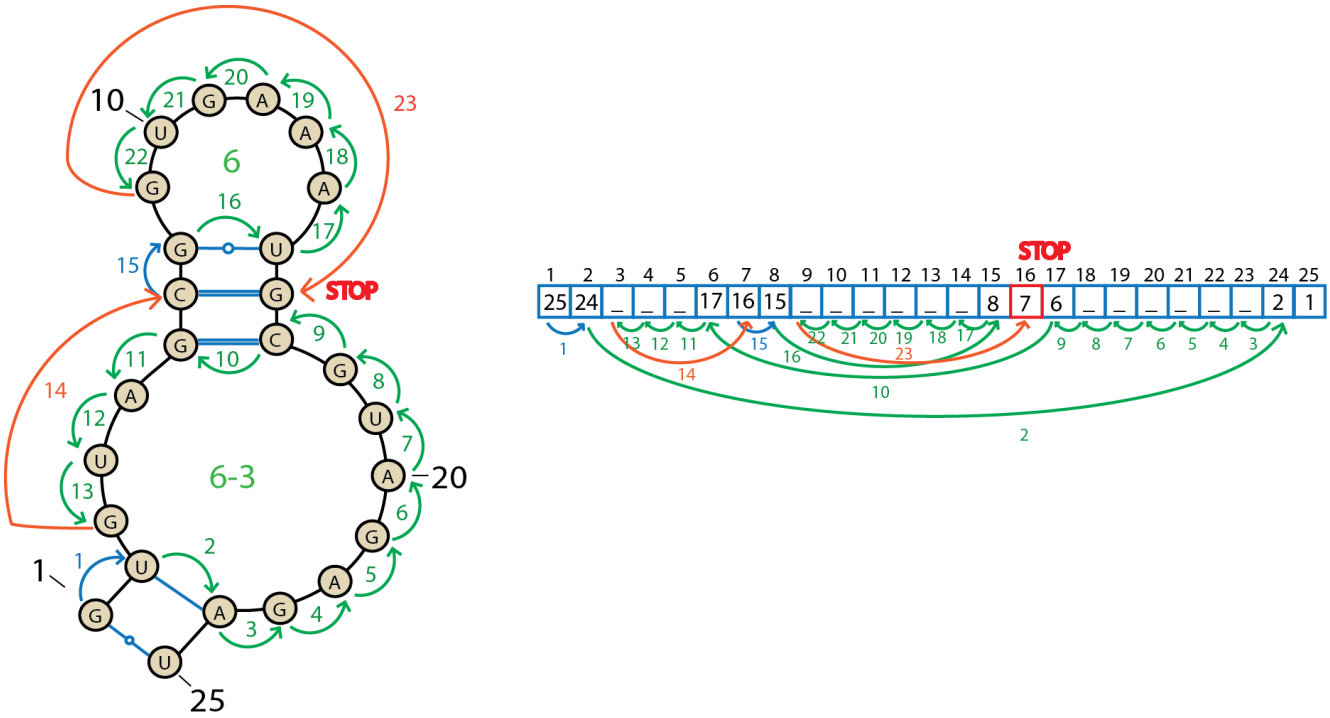


Figure 11: Scheme presenting the steps of the algorithm that finds and classifies the motifs in the RNA structures. The numbers correspond to the step number. Green arrows correspond to the structural motif and blue to the helices. The orange lines denote the jumps after finishing the motif. The STOP sign indicates the position where the algorithm stops.

6 Trajectory analysis

If the case of a trajectory in which multiple RNA conformations have to be analyzed and classified, every frame is characterized in detail as previously described. The main output of the program is a set of `.csv` files listing all the pairs and motifs: helices, triplexes, pseudo-knots etc. and the number of the frames in which the considered structure was present, its topology and participating nucleotides. A detailed description of all the output files can be found in Section 7.

6.1 Clustering

To recognize and characterize the dynamics of the secondary structure of the RNA molecule, we have to cluster the detected secondary structure motifs. Clustering is parameterized with two user-defined parameters:

- `time_cutoff` that defines the minimal percentage of frames in which the motif has to be present to be classified.
- `margin` that defines the minimal percentage of similarities between the two motifs to belong to the common cluster.

The fist step is to remove rare motifs (leaving only the significant ones) through their filtering according to the number of frames the motif appeared in. Below one can find an exemplary list of motifs after removing rare motifs:

1. C13-G29-U28-G27-C15-A14-C13-
2. C2-G42-C41-G40-C4-G3-C2-
3. C8-G35-C34-U33-G32-C11-U10-G9-C8-
4. C1-G43-G42-C41-G3-C2-C1-
5. C11-G32-A31-A30-G29-U28-A14-C13-A12-C11-
6. G9-C34-U33-G32-C11-U10-G9-
7. A14-U28-G27-G26-C16-C15-A14-
8. C11-G32-A31-A30-G29-C13-A12-C11-
9. G9-C34-U33-G32-A31-A30-G29-C13-A12-C11-U10-G9-

Next, the motifs distance matrix is computed. The distance between motifs is the number of their common nucleotides. The order of the nucleotides is not taken into account:

motifs	1	2	3	4	5	6	7	8	9
1	6	0	0	0	4	0	4	2	2
2	0	6	0	4	0	0	0	0	0
3	0	0	8	0	2	6	0	2	6
4	0	4	0	6	0	0	0	0	0
5	4	0	2	0	9	2	2	7	7
6	0	0	6	0	2	6	0	2	6
7	4	0	0	0	2	0	6	0	0
8	2	0	2	0	7	2	0	7	7
9	2	0	6	0	7	6	0	7	11

Then, the partners for all motifs are denoted. The partner has to satisfy the distance requirement expressed by the equation shown below:

if T[i][j]>threshold*min(len(i), len(j))

threshold parameter is defined by the user.

For threshold = 0.6 the first list of friends looks like this:

1. 1 5 7
2. 2 4
3. 3 6 9
1. 1 5 7
2. 2 4
3. 3 6 9
4. 4 2 4. 4 2

5. 1 5 8 9
6. 6 3 9
7. 7 1
8. 8 5 9
9. 3 5 6 8 9

Next, the motif with the longest list of partners is incorporated as the first one to the first cluster number zero. The motifs used in the first cluster have to be crossed out from the rest. Then the second longer is chosen, the picked motifs are crossed out ant so on.

1. ~~1 5 7~~
2. 2 4
3. ~~3 6 9~~
4. 4 2

5. ~~1 5 8~~
6. ~~6 3 9~~
7. 7 1
8. ~~8 5 9~~
9. **~~3 5 6 8 9~~**

repeat step 3 as long as there are motifs in the list ..

Output cluster list:

cluster 0
8. C11-G32-A31-A30-G29-C13-A12-C11-
5. C11-G32-A31-A30-G29-U28-A14-C13-A12-C11-
6. G9-C34-U33-G32-C11-U10-G9-
3. C8-G35-C34-U33-G32-C11-U10-G9-C8-
9. G9-C34-U33-G32-A31-A30-G29-C13-A12-C11-U10-G9-

cluster 1
2. C2-G42-C41-G40-C4-G3-C2-
4. C1-G43-G42-C41-G3-C2-C1-

cluster 2
1. C13-G29-U28-G27-C15-A14-C13-
7. A14-U28-G27-G26-C16-C15-A14-

Additionally, the average clusters are returned. The list consists of the longest motif for every cluster and the two vectors of float values describing every nucleotide participating in the representative motif. The float vectors represent the average number of hydrogen bonds created by a nucleotide in both WC-WC pairs and other interactions. We hope to show the average secondary structure of the molecule within the tertiary structure description. While analyzing the whole output generated by the script, one can understand the complex description of the RNA structure in the dynamics.

6.2 Parameters

Parameters are stored in a dictionary that is an easy manageable structure set. The data are organized by keys; both the keys and their values can be of any type. Here, we describe the parameter dictionary `parms` which contains the keys that are text descriptors of the parameters:

- `Singe/Traj` – determines [Single/Traj] whether a trajectory or a single frame will be analyzed.
- `file_name` – input [file name] the name of the file for a single frame analysis mode.
- `file_dcd` – input [file name] the name of the trajectory file for the trajectory analysis mode.
- `chains_names` – [list] the list of chains names the analysis will be performed for. If empty the analysis will be performed for all chains. **All analyzed chains are treated as a single chain**.
- `first_frame` – the number of the first frame [int] to be analyzed in the trajectory.
- `last_frame` – the number of the last frame [int/-1] to be analyzed in the trajectory, if -1 is detected the program finds the number of the last frame on its own.
- `cutoff` – the cutoff for the distance [Å] measured between the C1' carbons of every nucleotide. For distances larger than `cutoff` the program does not search for hydrogen bonds.
- `cutoff_stacking` – the margin distance [Å] for the stacking interaction measured between the centers of mass.
- `h_bond_atom` – ["donor"/"hydorgen"] if the hydrogen bond distance should be computed between donor and acceptor or hydrogen bond and acceptor.
- `h_bond_l` – the maximal length [Å] of the hydrogen bond.
- `h_bond_angle` – the minimal angle [degrees] of the hydrogen bond.
- `vmd` – the binary parameter; when turned on [0/1] a VMD application will be opened and the input structure will be displayed. This is possible only when VMD is properly installed and added to PATH. Step-by-step procedure is detailed in the 7.1.2 subsection.
- `table_nucleotides` – the `.csv` file [file name] determining the hydrogen donors, acceptors and nucleotide edges for every nucleotide. By editing this file, one can remove certain interactions from the analysis.

- `table_charges` – the `.csv` file [file name] determining the electric charges, Van der Waals radii and depths of Lennard-Jones potential for atoms in nucleotides. These parameters are given for two main force fields: AMBER and CHARMM, but there is a column MY_OWN for the user to put other parameters if needed.
- `force_field` – the name of the force field (AMBER/CHARMM/MY_OWN) to be used by the program while computing stacking energies.
- `vdw_cutoff_stacking` – maximal value [-0.5] of the Van der Waals energy for stacking interaction. If the VdW energy is smaller than given `vdw_cutoff_stacking` parameter the stacking interaction is detected.
- `margin` – the minimal percentage [(0.0,1.0)] of nucleotides that have to be common for both motifs to belong to the same cluster.
- `time_cutoff` – the minimal percentage [(0.0,1.0)] of the frames that the motifs must be present in in order to be incorporated into the cluster analysis.
- `max_memory_GB` – the maximal memory [int GB] the single thread is using at once.
- `threads` – the number [int] of CPUs to be used while analyzing the trajectory.
- `MINT_home` – path to the directory where MINT is stored on the computer - this enables running MINT from different localizations than its directory.
- `only_analysis` – [True/False] specifies the program instead of running the whole analysis reads in the pickles of performed previously analysis, performs computations for the missing frames and creates output files. You can run with this parameter turned on when your computations were disturbed for some reason. In this mode MINT uses only one CPU.
- `pdb_list` – [file name] if not empty, MINT will read in the list of PDB ids (a file with a PDB ids put in a column - one per row), download the structure from PDB database, unpack, protonate and performed analysis in a single frame mode. Every analysis will be located in the separate directories automatically.

7 Output files

The script generates multiple files both in the single frame analysis mode and in the trajectory mode. The generated files name begin with the `file_name` for the single frame mode and both `file_name` and `file_dcd` for the trajectory mode. They all are created in the trajectory where input files are placed.

- `_description` – contains a complete description of the structure for every frame. The exemplary description is shown in the end of this manual. The single frame description contains a complete list of used parameters, list of helices, motifs, triplexes, pseudo-knots. One can also find a list of both WCWC and non-canonical interactions, with the exact parameters of hydrogen bonds and stacking interactions. Additionally, there is a dot-bracket representation of the secondary structure that can be used for visualization or energy computation. The frames are separated by the headers: `frame number`.

- _MINT.xls – a complete .xls file collecting all of the below .csv files. For every .csv file a separate sheet is created.

- _pairs_in_time.csv – csv file containing all pairs that appeared during the trajectory. The file contains the following columns: number of first nuc, pair_nucleotides (the numbering of paired nucleotides), pair type (e.g. WCWC) , pair configuration (cis or trans), vmd, the number of frames a pair was present, frame numbers when a pair was present, the exemplary data record is shown below:

number of first nuc	pair _nucleotides	pair type	pair configuration	vmd	number of frames pair was present	frames when pair was present
515	G515/C536	WC/WC	Cis	resid 515 536	100,00%	0 → 1
516	U516/A533	WC/Hoogsteen	Trans	resid 516 533	50,00%	0
522	C522/G527	WC/WC	Cis	resid 522 527	100,00%	0 → 1

- motifs_in_time.csv – several csv files are created, separate for every type of the structure. Csv files are easy to manipulate and can be opened with any popular spread-sheet applications. All of the files:

  – _helices_in_time.csv,
  – _motifs_in_time.csv,
  – _pseudo_in_time.csv,
  – _triplex_in_time.csv,

  contain four columns:

  – motif_topology - absent in the _helices_in_time.csv,
  – motif_nucleotides - nucleotides creating the motif,
  – vmd - ready to pasted into the VMD residual identifiers,
  – the number of frames a helix was present,
  – frame numbers in which a helix was present.

  The example can be found in the table below (residues are removed just for presentation) :

- _nucleotides_characteristics.csv – csv file with every nucleotide represented in the separate line. Next, every field contains the nucleotide, the number of hydrogen bonds and the number of frames in which this pair was present. E.g output:

motif _topology	motif_nucleotides	vmd	percentage of frames motif was present	frames when motif was present
7-5	G515-C536-...-U516-G515-	resid 515 536 .. 516 515	100,00%	$0 \rightarrow 1$
4	C522-G527-...-A523-C522-	resid 522 527 .. 523 522	100,00%	$0 \rightarrow 1$
0-6	C504-G541-..-G505-C504-	resid 504 541 .. 505 504	100,00%	$0 \rightarrow 1$

G517	517	1-G529:4		
A502	502	2-U543:10		
G515	515	3-C536:8	4-C536:1	2-C536:1

indicates that the nucleotide A502 paired with U543 with 2 hydrogen bonds for 10 frames.

- `_nucleotides_eval.csv` – csv file contains the physical description of the single nucleotide. The `2d-hbonds` correspond to the number of hydrogen bonds created by the nucleotide in the WC pairs, analogously `3d-hbonds` is the hydrogen bonds number in non-WC pairs. In the case of the trajectory it these are average numbers of bonds. The `Coulomb` column contains the energy of the Coulomb interaction, the same with next columns: `Vdw` and the `sum`. These interactions are originally computed for pairs - a single nucleotide is described with the sum of all interactions of a given kind. Therefore, if one is looking on how much the nucleotide is fisted this is a good measurement, but while looking on the several nucleotides one have to keep in mind not to incorporate energies more than once. A fragment of the example output can be seen below:

nuc	num	2d-hbonds	3d-hbonds	Coulomb	Vdw	sum
A502	502	2,00	0,00	1,57	-15,74	-14,18
G517	517	0,00	0,50	5,73	-10,65	-4,92
G515	515	3,00	0,00	2,19	-20,83	-18,64

- `_motifs_clusters.csv` – a csv file with clusters of motifs - it contains all of the motifs assigned to the clusters, and overall percentage of the frames the given motif was present and the frames numbers. Frames and clusters are numbered from zero. File is presented below:

cluster 0	4-0	A520-A533-...- G521-A520-	resid 520 533 ... 521 520	0.3	$3 \rightarrow 49$
cluster 0	7-5	G515-...-U516-G515-	resid 515 536 ... 516 515	0.7	$0 \rightarrow 25 \rightarrow 8$
cluster 0	2-4	G515-...-U516-G515-	resid 515 536 ... 516 515	0.3	$3 \rightarrow 49$
cluster 1	4	C522-G527-...-A523-C522-	resid 522 527... 523 522	1.0	$0 \rightarrow 9$
cluster 2	0-6	C504-G541-...-G505-C504-	resid 504 541 ... 505 504	1.0	$0 \rightarrow 9$

- `_average_motifs.csv` - a csv file representing the list of average motifs, driven from the cluster list and nucleotide characteristics list. The vector described as a `2D` is the average number of hydrogen bonds created by the above named nucleotide in the WCWC interactions, a `3D` vector is analogous but for non-canonical interactions.

1	4	C522-G527-...-A523-C522-	resid 522 ... 522	1.0	$0 \rightarrow 9$
	2D	3.0 3.0 2.8 ... 0.0 3.0			
	3D	0.0 3.0 0.8 ... 2.1 0.0			
2	0-6	C504-G541-...-G505-C504-	resid 504 ... 504	1.0	$0 \rightarrow 9$
	2D	2.9 2.9 ... 2.8 2.9			
	3D	0.0 0.0 ... 0.0 0.0			

- Files needed for visualization using external tools:
  - `_RNAStructML.xml`
  - `_varna.html`
  - `_2D.pdb`
  - `_3D.pdb`
  - `_coulomb.pdb`
  - `_VDW.pdb`
  - `_stacking_sum.pdb`
  - `vmd_run.tcl`

Detailed description of the visualization procedures can be found below.

7.1 Visualization

MINT enables many different ways of data visualization. The user can display a colored input .pdb structure according to several parameters (motifs, secondary and tertiary contacts, electrostatic, VDW energies) both in the `Single` and `Trajectory` modes. The same parameters are colored on the secondary structure.

For visualization it is preferable to install the following programs:

- VARNA for secondary structure visualization and coloring [1].
- VMD for tertiary structure visualization [7].
- RNAMovies for secondary structure trajectory visualization (`http://bibiserv.techfak.uni-bielefeld.de/rnamovies/`).

7.1.1 VARNA

The VARNA [1] program produces the interactive image of the secondary structure from the given sequence and a dot-bracket representation of the given single-stranded RNA molecule. The sequence is retrieved directly from an input .pdb file. The secondary structure is computed from the list of Watson-Crick base pairs – in the case of the single frame from an input .pdb file and in the case of the trajectory mode from the list of the most represented WC pairs.

VARNA is also available in the form of the java applet for submitting into the html website. Using the applet it is easy to color a secondary structure using any set of numbers – in our case these are:

- Number of Watson-Crick hydrogen bonds .
- Number of non-Watson-Crick hydrogen bonds.
- Coulomb term of Stacking energy.
- VDW term of Stacking energy.
- Sum of the coulomb and Van der Waals interactions.

In case of all energies (VDW, Coulomb and Van der Waals) the scale of colors is reversed, therefore the red nucleotides are the ones that are influenced by the strongest hydrogen bonding, VDW and Coulomb interactions.

To launch the VARNA applets simply open the `_varna.html` file with your favorite browser with enabled java. Figure 7.1.1 presents the output from the analysis of the exemplary structure and its trajectory.

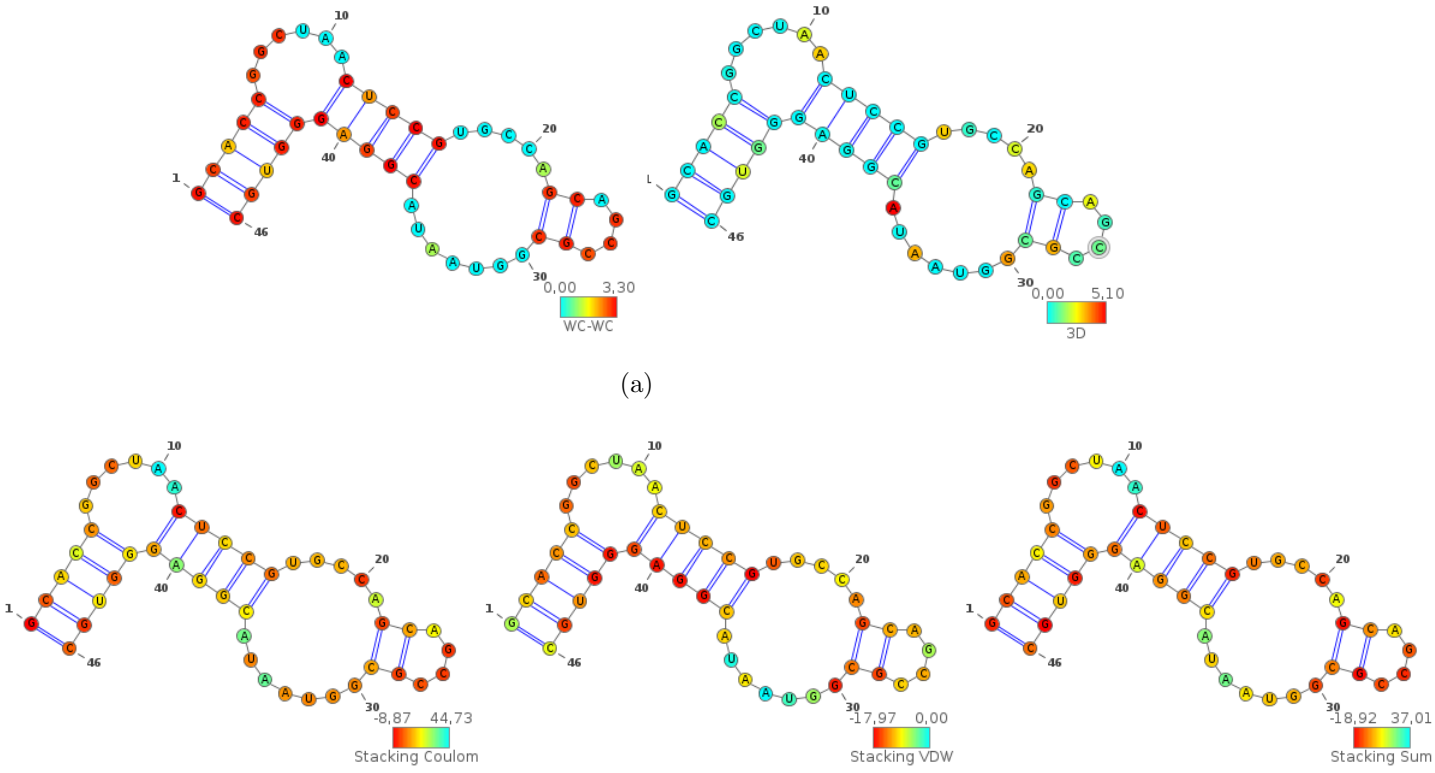


Figure 12: Created with VARNA output of MINT.

7.1.2 VMD

Representing motifs on the tertiary structure works through creating representations of the regions of the nucleic acid molecule in the input .pdb file using the VMD program [7]. In the `VMD` one can represent fragments on molecule using different representations and color, all can be managed from `Graphical Representations` menu (`Graphics > Representations`). The `vmd_tcl` script loads the structure and creates `Reps` for every motif and helix in the structure what results with a view of the three-dimensional molecule colored accordingly to the detected structural components:

- helices: yellow (vmd color code:4)
- pseudo-knots: tan (vmd color code:5)
- triplexes: silver (vmd color code:6)
- loops: green (vmd color code:7)

If the `VMD` program is in your PATH you can turn on the `vmd` parameter and it will launch automatically. Otherwise, you can do it manually by going into your MINT inputs/outputs catalog and typing in the terminal:

```
your_vmd_location\vmd -e vmd_run.tcl
```

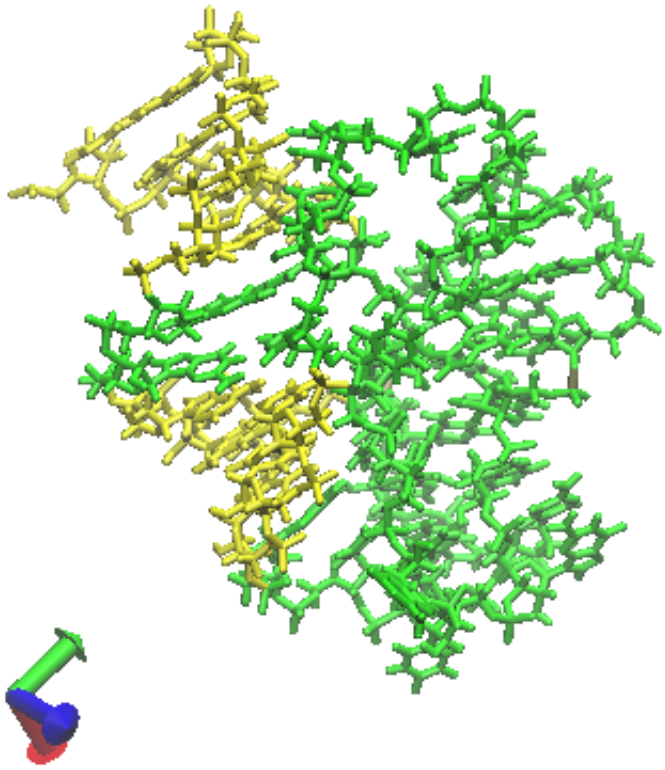


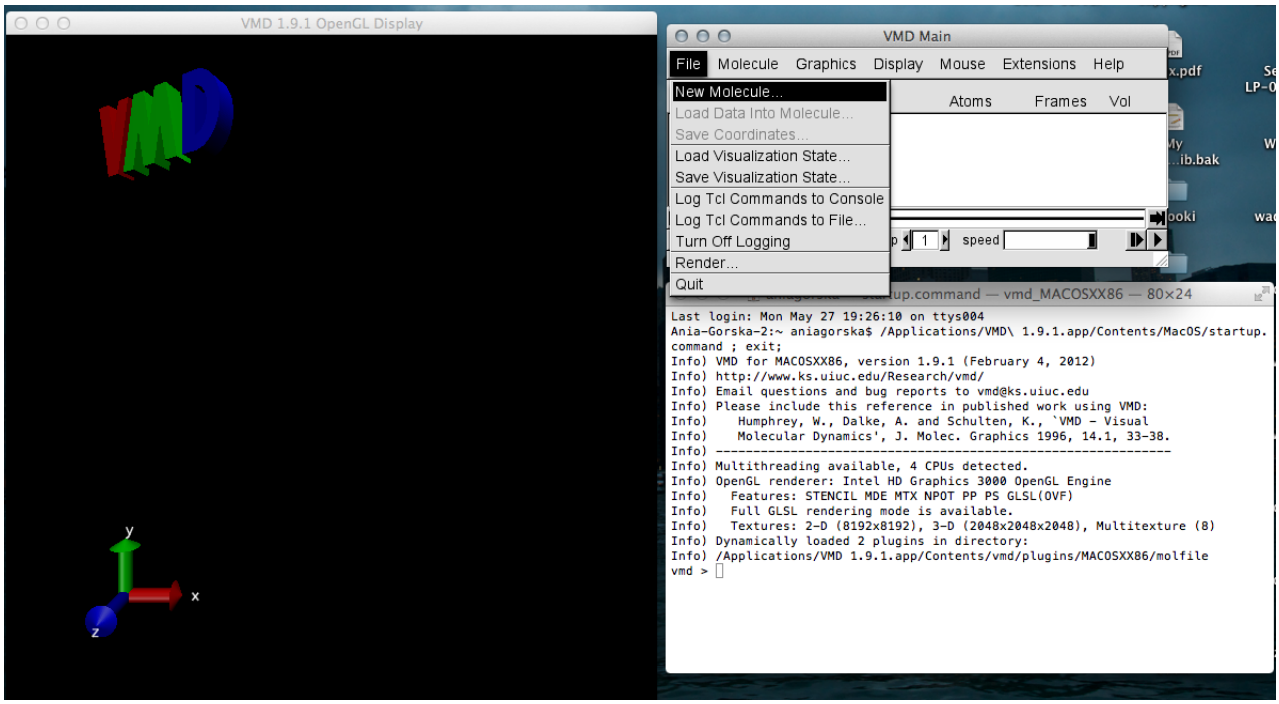
Figure 13: Result of running the exemplary structure `vmd_run.tcl` script in the VMD program after changing display into `Orthographic` and changing the background color.

Representing hydrogen bonding and stacking on the tertiary structure The program produces several .pdb files with occupancy column replaced with the values of:

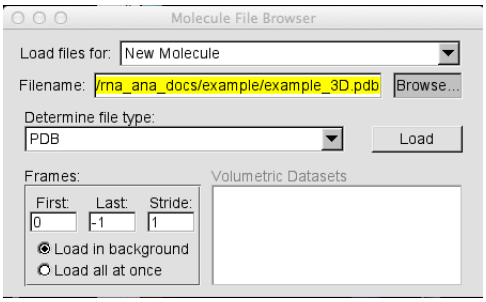
- `_2D.pdb` – the number of hydrogen bonds in the Watson-Crick pairs created by the given nucleotide.
- `_3D.pdb` – the number of hydrogen bonds in the non Watson-Crick pairs created by the given nucleotide.
- `_coulomb.pdb` – the value of the Coulomb energy for a given nucleotide (the sum of all interactions the nucleotide is involved in).
- `_VDW.pdb` – the value of the Van der Waals energy for a given nucleotide (the sum of all interactions the nucleotide is involved in).
- `_stacking_sum.pdb` – the sum of Van der Waals and Coulomb energies.

In the case of trajectory these .pdb files contain the average values for the analyzed trajectory. `VMD` is a powerful tool that has a complete user guide and tutorial that can be found on the VMDs home website (`http://www.ks.uiuc.edu/Research/vmd/`). Here, we describe a short manual on how to visualize the computed data.

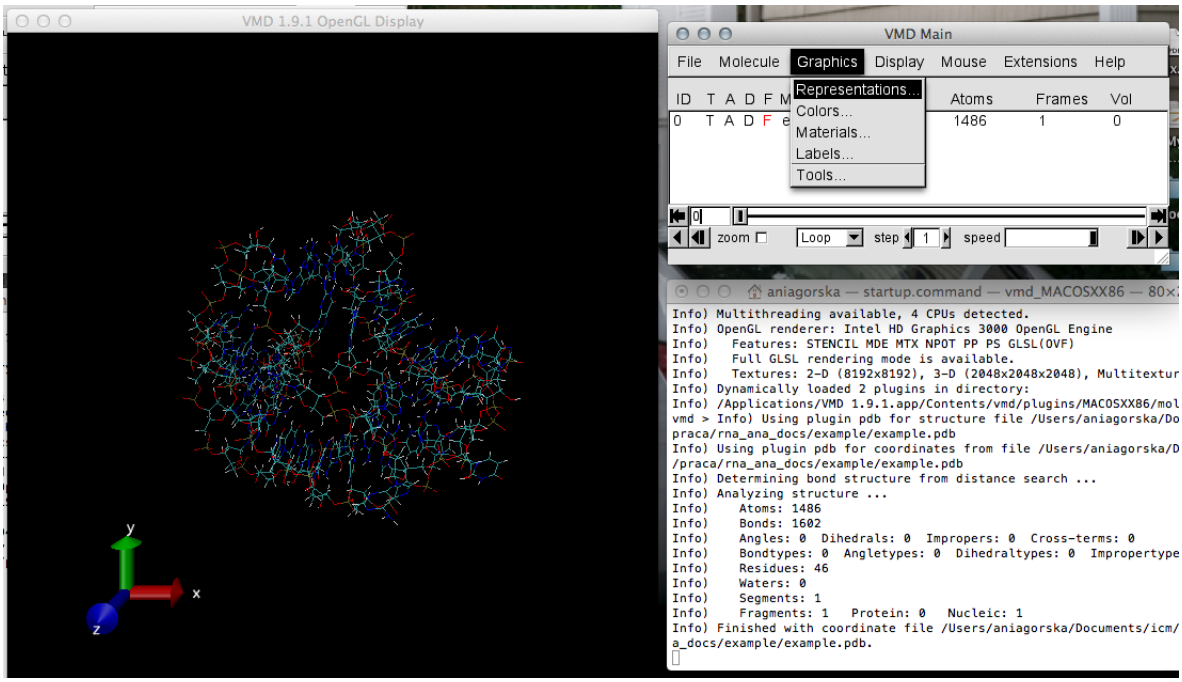
The user has to open the `VMD` program and load one of the .pdb files, through choosing the `New Molecule` from the `File` drop-down menu. As it is shown below:



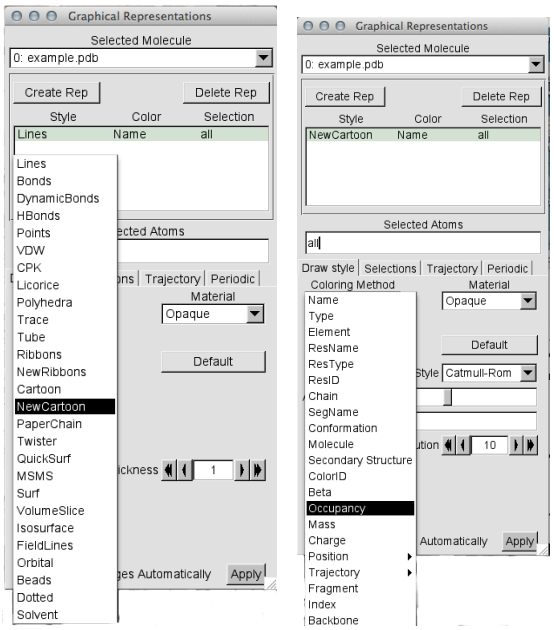
Next, one has to `Browse` and `Load` a desired .pdb file:



Having the loaded structure, one needs to open the `Representations` window from the `Graphics` drop-down menu:

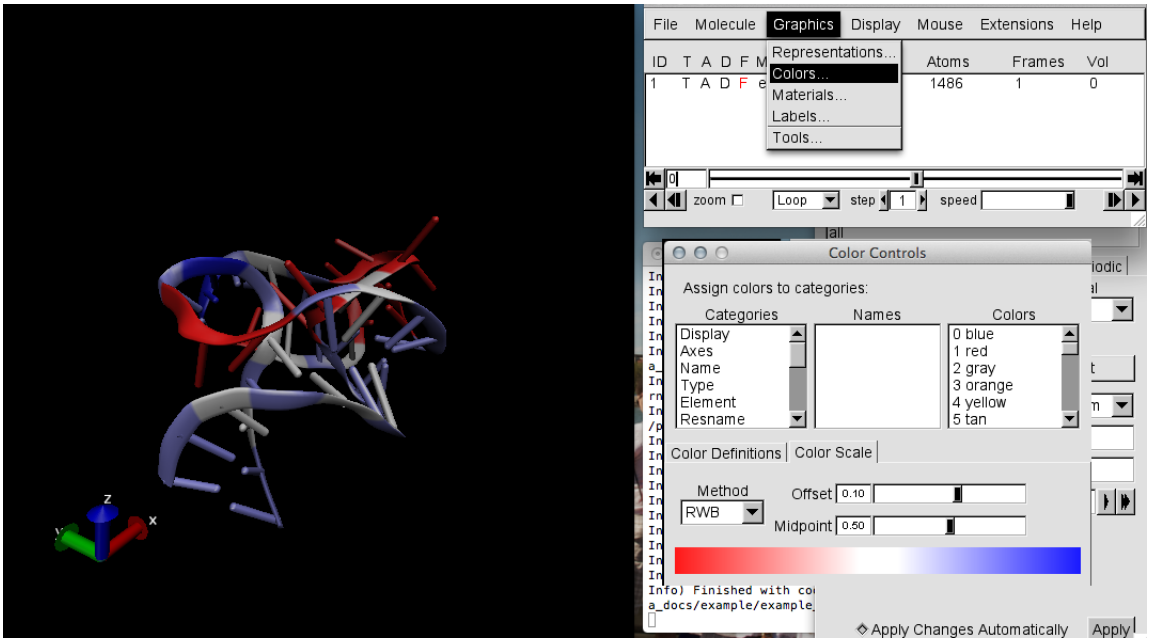


The `Representations` menu allows one to create multiple different representations. To color the structure by the occupancy column from the .pdb file, we propose to change the `Drawing Methods` into the `New Cartoon` and the `Coloring Method` into the `Occupancy`:



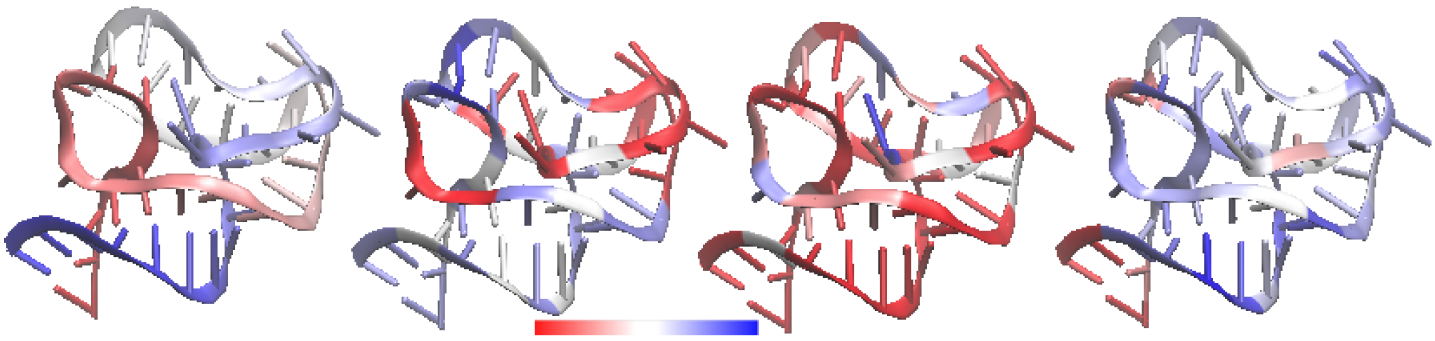
(a) (b)

Then the color scale can be altered through `Graphics` > `Color` > `Color Scale`:



If you proceed int the same way with more than one .pdb structure you can use the `Move` tool (`Mouse` > `Move` > `Molecule`) and view the colored structures at the same time as it is presented in Figure 7.1.2.

Secondary structure trajectory can be visualized with the RNAMovies [4] program. The MINT returns the `RNAStructML.xml` containing the trajectory of the secondary structure. For



a) number of hydrogen bonds in WC/WC interactions.

b) number of hydrogen bonds in non WC/WC interactions.

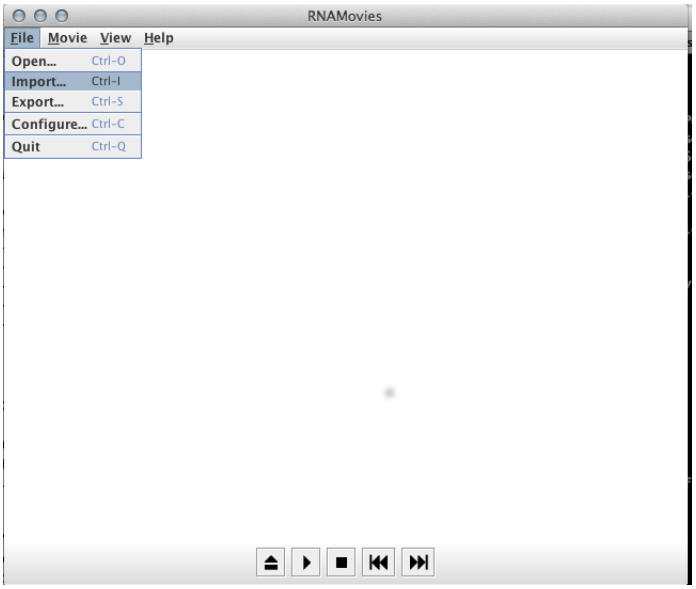
c) sum of VDW interactions per nucleotide.

d) sum of electrostatic interactions per nucleotide.

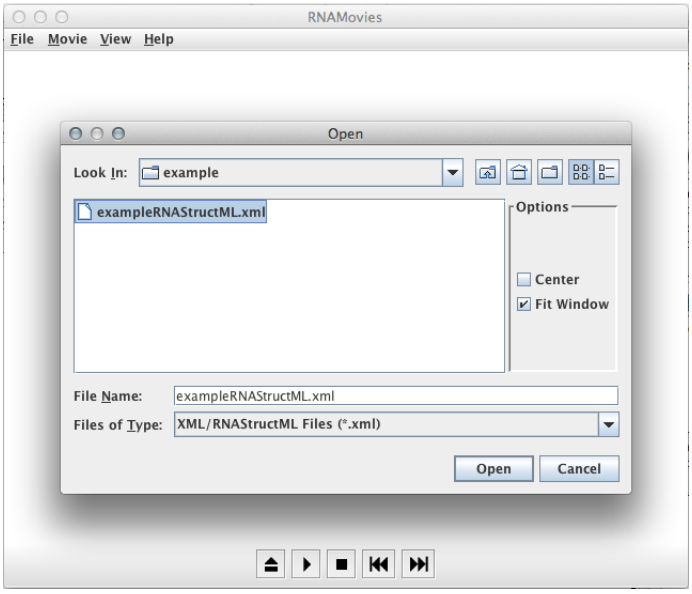
every frame the dot-bracket representation is written into the .xml file. This allows to produce the movie of the secondary structure trajectory.

The RNAMovies [4] java .jar file can be downloaded from its home page: `http://bibiserv.techfak.uni-bielefeld.de/rnamovies/`. Here we present a short tutorial on how to open a .xml file and view the trajectory.

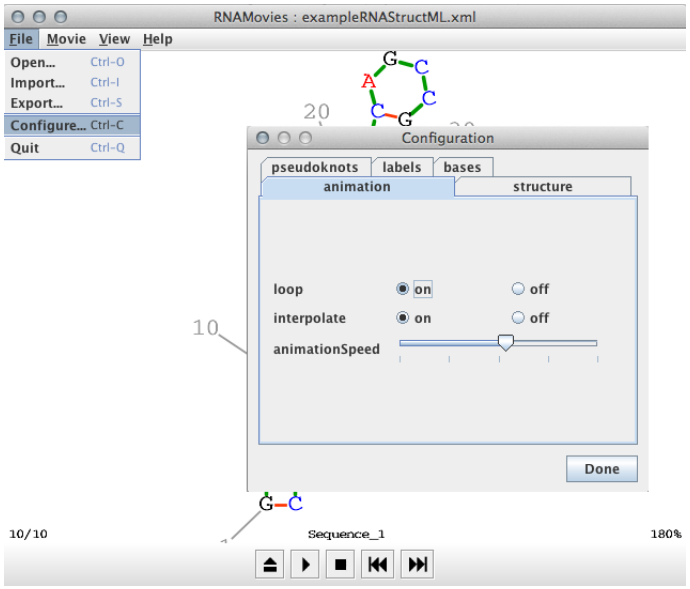
First, one has to open the RNAMovies and choose the `Import` option from the `File` drop-down menu.



Find a proper .xml file in your disk, and click `Open`:



One can navigate the trajectory using the arrows in the bottom of the window. If you want to loop the trajectory or change the pace go to the `File` > `Configure` menu:



The animated trajectory can be written into the .gif file (`File` > `Export`).

7.2 Correlations

In the `MINT` directory user can find additional python script `CORRELATIONS.py` computing `phi coefficient` for every pair of nucleotides in the structure. The `phi coefficient` is computed using the equation:

$$\phi = \frac{n_{11}n_{00}}{\sqrt{n_{\bullet 1}n_{\bullet 0}n_{0\bullet}n_{1\bullet}}}$$

where:

- n_{11} is a number of frames when both nucleotides are creating WCWC pair.
- n_{00} is a number of frames when none of the nucleotides is creating WCWC pair.
- n_{01} is a number of frames when first of the nucleotides is creating WCWC pair and the second nucleotide is not, analogously n_{10} is the number of frames when first nucleotide is paired and second is free.

and:

- $n_{\bullet 1} = n_{11} + n_{01}$
- $n_{1\bullet} = n_{11} + n_{10}$
- $n_{\bullet 0} = n_{00} + n_{10}$
- $n_{0\bullet} = n_{00} + n_{01}$

`Phi coefficient` takes values between -1 and 1. It is believed that when coefficient value is close to 0 the correlation is not reliable. In the heat map, all values larger then `cutoff` will be marked red, all lower than `-1*cutoff` are marked blue. The rest will remain white. The level of `cutoff` is defined by the user.

The script produces a matrix that is both written into the text file and printed in the form of the heat map. Figure 7.2 presents heat map of `phi coefficient` for the previously discussed exemplary RNA molecule and its 10 frame molecular dynamics.

`CORRELATIONS.py` script uses as an input the `MINT` output: `pairs_in_time.csv` type, cutoff and list of numbers of nucleotides you want to compute the coefficient for:

```
 python pairs_correlations.py step6_pairs_in_time.csv 0.4 "[(1210,1220),(985,995),
(1043,1047)]"
```

The list of nucleotides has to be specified in the square brackets and quotation marks. Inside round brackets the ranges of nucleotides must be specified.

If the user wants to compute the phi coefficient for all nucleotides should not specify list of nucleotides like:

```
python pairs_correlations.py step6_pairs_in_time.csv 0.4
```

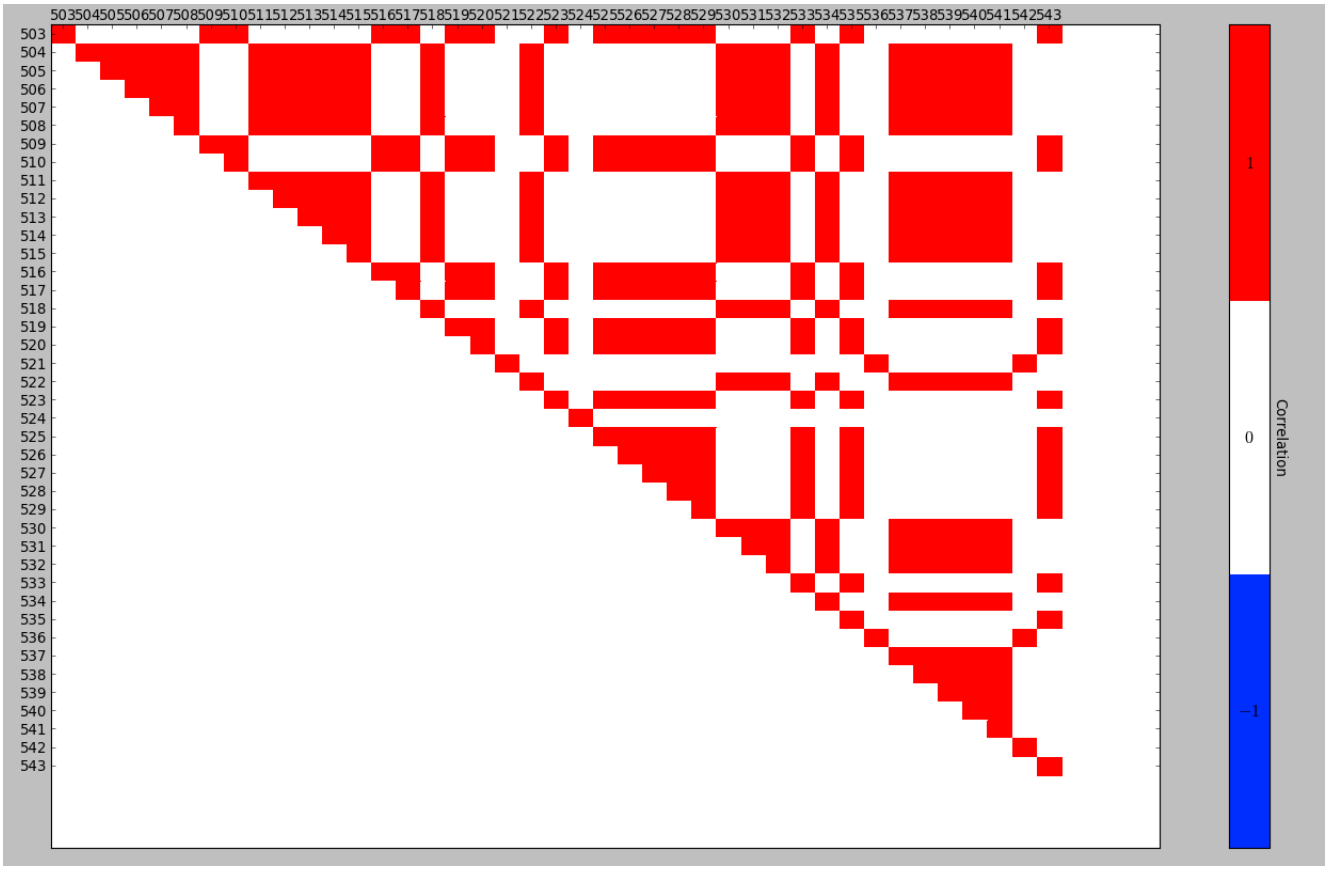


Figure 14: Heap map of `phi coefficient` for the the same exemplary molecule as shown before, and for discussed trajectory. These map was calculated with 0.4 `cutoff`.

Heat map is only printed for the half of the matix, another half would be identical. All of the nucleotides are correlated with themselves - the diagonal is colored red. All of the nucleotides creating WCWC pairs with each other are also positively correlated. The neighboring pairs that work together will also be visible as correlated in all to all manner. The negative correlation suggests that there are pairs that opens when another one closes.

8 Appendices

.1 Adding hydrogens to a .pdb structure

Following the Leontis and Westhof [8] classification of interacting nucleotides, our program searches for hydrogen bonds in the given structure. In order to properly recognize all the hydrogen bonds, the hydrogen atoms have to be present in the input .pdb file. If one analyzes the files from MD simulations, the hydrogen atoms were added to the structure during the preparation of an MD run, according to the atom type definitions in the force field. Various tools of the MD packages can be used to assign the positions of hydrogens (for example Xleap from AmberTools).

However, if a structure from the PDB database has to be analyzed and the user does not have experience with MD methods, hydrogen atoms can be added using on-line servers. We have tested

a few of them and we can recommend the Molprobility service [3] (`http://molprobity.biochem.duke.edu/`). It works even with the structures as large as the ribosomal subunits in an acceptable time span. What is more, the software can be downloaded from the `http://kinemage.biochem.duke.edu/software/reduce.php` website and used offline.

References

[1] Guillaume Blin, Alain Denise, Serge Dulucq, Claire Herrbach, and Hélène Touzet. Alignments of RNA structures. *IEEE/ACM transactions on computational biology and bioinformatics / IEEE, ACM*, 7(2):309–22, 2009.

[2] Victor A. Bloomfield, Donald M. Crothres, and Ignactio Jr. Tinoco. *Nucleic Acids: Structures, Properties, and Functions*. University Science Books, 1999.

[3] Vincent B Chen, W Bryan Arendall, Jeffrey J Headd, Daniel a Keedy, Robert M Immormino, Gary J Kapral, Laura W Murray, Jane S Richardson, and David C Richardson. MolProbity: all-atom structure validation for macromolecular crystallography. *Acta crystallographica. Section D, Biological crystallography*, 66(Pt 1):12–21, January 2010.

[4] Dirk Evers and Robert Giegerich. RNA Movies : visualizing RNA secondary structure spaces. *Bioinformatics (Oxford, England)*, 15(1):32–37, 1999.

[5] Fonseca Guerra, F Matthias Bickelhaupt, Jaap G Snijders, De Boelelaan, Nl-H V Amsterdam, and Theoretische Chemie. Hydrogen Bonding in DNA Base Pairs : Reconciliation of Theory and Experiment. *Journal of the American Chemical Society*, (122):4117–4128, 2000.

[6] Pavel Hobza. Stacking interactions. *Physical Chemistry Chemical Physics*, 10(19):2595–2610, May 2008.

[7] W Humphrey, A Dalke, and K Schulten. VMD: visual molecular dynamics. *Journal of Molecular Graphics*, 14(1):33–8, 27–8, February 1996.

[8] Neocles B Leontis, Jesse Stombaugh, and Eric Westhof. The non-Watson-Crick base pairs and their associated isostericity matrices. *Nucleic acids research*, 30(16):3497–531, August 2002.

[9] A Lescoute and Eric Westhof. The interaction networks of structured RNAs. *Nucleic Acids Research*, 34(22):6587–604, January 2006.

[10] Jerzy Leszczynski, Pavel Hobza, and J Heyrovsky. S Electronic Properties , Hydrogen Bonding , Stacking , and Cation Binding of DNA and RNA Bases. *Biopolymers*, 61:3–31, 2002.

[11] Naveen Michaud-Agrawal, Elizabeth J. Denning, Thomas B. Woolf, and Oliver Beckstein. MDAnalysis: A Toolkit for the Analysis of Molecular Dyanmic Simulations. *Journal of Computational Chemistry*, 2012.

[12] James C Phillips, Rosemary Braun, Wei Wang, James Gumbart, Emad Tajkhorshid, Elizabeth Villa, Christophe Chipot, Robert D Skeel, Laxmikant Kalé, and Klaus Schulten. Scalable molecular dynamics with NAMD. *Journal of Computational Chemistry*, 26(16):1781–802, December 2005.